[1]Dr. D. R. V. A. Sharath Kumar

[2]K. Narasimha Rao

# Low Power-Area Implementation of 4-Bit ALU in Cadence Virtuoso Platform

Journal of Electrical Systems

***Abstract:*** - The 4-bit ALU designed using Cadence is a arithmetic logic unit capable of performing various operations on 4-bit binary numbers. It takes in two 4-bit inputs, A and B, and producesa4-bitoutput, F, along with carry-out and overflow flags. The design incorporates a combination of combinatorial and sequential logic circuits to achieve the desired functionality. The ALU supports basic arithmetic operations like addition and subtraction, as well as logical operations such as AND, OR, XNOR and XOR. It utilizes multiplexers, adders, Code Converters and logic gates to perform the different operations. The inputs are decoded to determine the specific operation to be executed, and the output is generated accordingly. The 4-bit ALU designed using Cadence is a versatile and optimized circuit capable of performing various arithmetic and logical operations on 4-bit binary numbers. Its efficient design and use of advanced techniques ensure high performance and reliability. The Cadence tool set enables the seamless development and simulation of the ALU design

***Keywords:*** performance, reliability, operations, ALU

## I. INTRODUCTION

A low power area 4-bit ALU is a circuit that performs arithmetic and logical operations on 4-bit binary numbers by consuming minimal power and occupying a small area on a chip. An arithmetic logic unit (ALU) is a digital circuit used to perform arithmetic (addition, subtraction, multiplication, etc. ) and logic(XOR, XNOR, NAND, NOR etc.) operations. It represents the fundamental building block of the central processing unit (CPU) of a computer. Modern CPUs contain very powerful and complex ALUs. Cadence is a popular Electronic Design Automation (EDA) software suite that allows engineers to design and verify integrated circuits. Using Cadence, designers can create and optimize the layout of the ALU, ensuring that it meets the low power and area requirements. The software provides various tools and features to help designers reduce power consumption, such as power gating and voltage scaling techniques. These techniques can be implemented through the use of specialized cells and libraries provided by Cadence. Additionally, Cadence offers simulation and verification capabilities, allowing designers to test the functionality and performance of the ALU before fabrication. This helps in identifying any potential issues or design flaws early in the development process, saving time and resources. Efficiently design and optimize a low power area 4-bit ALU, ensuring that it meets the desired power, area, and performance specifications. One of the key techniques in low power design is power gating. Power gating involves selectively turning off power to specific parts of the circuit when they are not in use, reducing static power consumption. By strategically implementing power gating in the 4-bit ALU design, we can effectively manage power consumption and enhance overall efficiency. Clock gating is another essential technique in low power design. It involves controlling the clock signal to specific parts of the circuit, allowing them to operate only when needed. By incorporating clock gating mechanisms in the ALU design, we can further reduce dynamic power consumption and improve power efficiency. At the transistor level, designers can optimize the design by carefully selecting transistor sizes, configurations, and layout to minimize power leakage and enhance performance. By fine-tuning the transistor-level details of the 4-bit ALU, we can achieve a balance between functionality, power efficiency, and area optimization. Routing paths play a crucial role in ensuring efficient signal propagation within the ALU. By designing optimized routing paths that minimize signal delays and power consumption, we can enhance the overall performance of the ALU while meeting the low power area targets. When working on a 4-bit ALU design, the limited bit width poses a unique challenge. Designers must carefully allocate resources, optimize data paths, and control logic to ensure efficient operation within the

[1] *Associate Professor, Maturi Venkata Subba Rao(MVSR) Engineering College, acharyasharath79@gmail.com

2* Assistant Professor, Maturi Venkata Subba Rao(MVSR) Engineering College, narasimharao_ece@mvsrec.edu.in

constraints of a 4-bit architecture. Through thoughtful design decisions and strategic utilization of resources, we can create a high-performance 4-bit ALU that excels in both power efficiency and area optimization.

## II.LITERATURE REVIEW

Numerous studies have delved into the realm of low power design techniques for ALUs, focusing on strategies to minimize power consumption while optimizing the use of space. Power gating and clock gating have emerged as prominent techniques in the literature, with researchers emphasizing their effectiveness in reducing power consumption in ALU designs. In the context of Cadence Virtuoso, researchers have highlighted the significance of leveraging the platform's advanced features to implement low power area designs efficiently.

By harnessing the tools and capabilities offered by Cadence Virtuoso, designers can navigate the complexities of optimizing power efficiency and area utilization in their 4-bit ALU implementations. Transistor-level optimization has also been a focal point in the literature, with studies emphasizing the importance of carefully selecting transistor configurations, sizes, and layouts to minimize power leakage and enhance performance. Researchers have explored various transistor-level design strategies to achieve a balance between functionality, power efficiency, and area optimization in 4-bit ALU implementations. Routing path optimization has been another key area of interest in the literature, with researchers investigating techniques to design efficient signal propagation paths within the ALU. By optimizing routing paths to minimize signal delays and power consumption, designers can enhance the overall performance of the ALU while meeting low power area targets. The literature surrounding 4-bit ALU designs has underscored the challenges posed by the limited bit width of such architectures. Researchers have explored methods to allocate resources efficiently, optimize data paths, and control logic within the constraints of a 4-bit architecture. Through meticulous design decisions and resource utilization, researchers aim to create high-performance 4-bit ALUs that excel in power efficiency and area optimization. Overall, the literature review on the low power area implementation of a 4-bit ALU in Cadence Virtuoso showcases a rich tapestry of research findings, methodologies, and design strategies. By synthesizing insights from existing studies and pushing the boundaries of innovation, designers can continue to advance the field of low power area design in 4-bit ALU implementations.

## III.BLOCK DIAGRAM

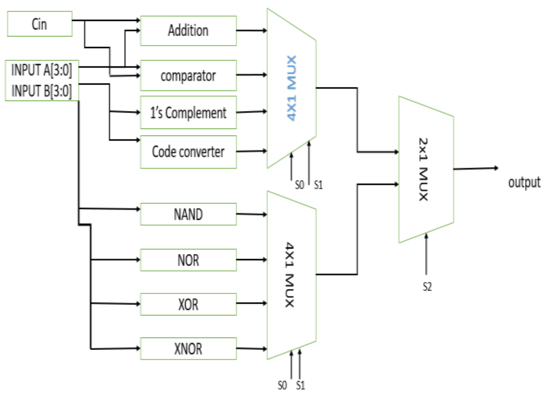


**Fig.1** Block diagram of ALU

## IV. ARCHITECTURE AND IMPLEMENTATION

### NOR GATE:

It contains 3 transistors .2 NMOS and 1 PMOS . When both sources are at logic LOW, the output is LOW. When any input is at logic LOW, the output is LOW. Only when both inputs are LOW is the output HIGH.

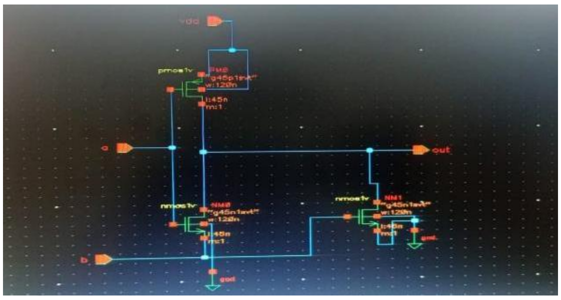

**Fig.2** : Schematic of Nor Gate

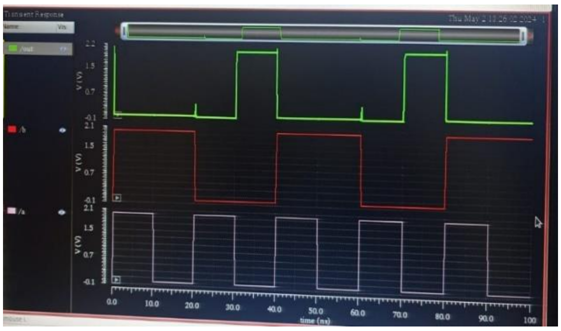

**Fig.3**: Waveform of Nor gate

**NAND GATE:**

It consists of 4 Transistors .2 PMOS and 2 NMOS A two-input NAND gate is a digital combination logic circuit that performs the logical inverse of an AND gate. While an AND gate outputs a logical “1” only if both inputs are logical “1,” a NAND gate outputs a logical “0” for this same combination of inputs.

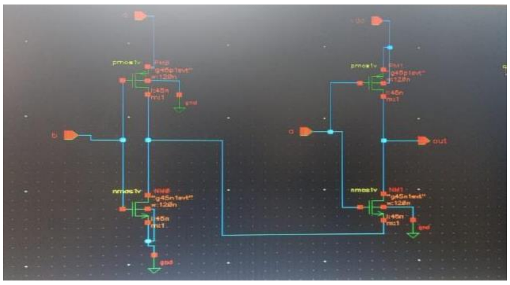

**Fig.4:** Schematic of NAND gate

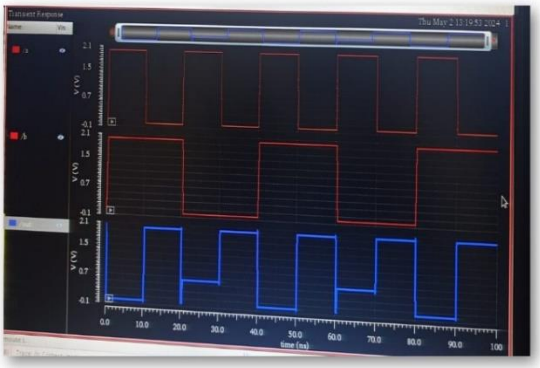

**Fig .5:** Waveform of Nand gate

**XOR & XNOR GATE:**

XOR and XNOR gates are implemented by using 6 Transistors.3 are PMOS and 3 are NMOS. For XOR, If both inputs are “0” (same inputs), the output is “0”. When one input is “0” and the other is “1” (different inputs), the output is “1”. When both inputs are “1” (same inputs), the output is “0”. For XNOR, In XNOR gate the Output is high (1) when both inputs are same (either both 0 or both 1), and low (0) when the inputs are different.

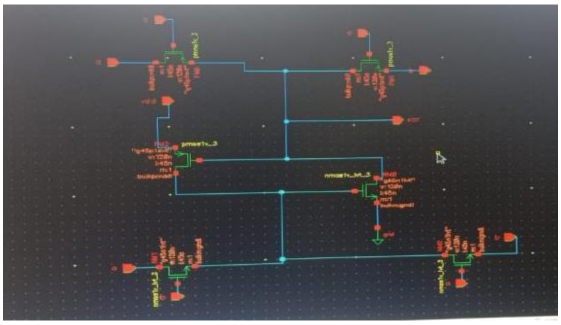

**Fig.5:** Schematic of XOR and XNOR gate

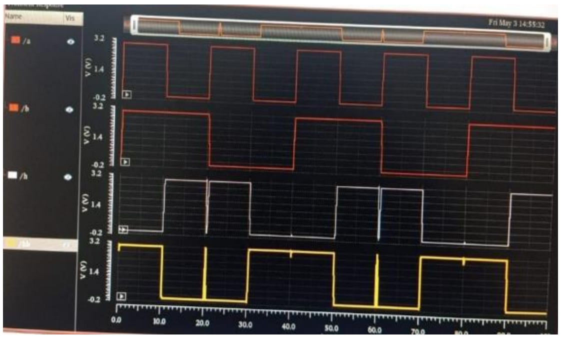

**Fig.6:** Waveform of XOR and XNOR gate

**FULL ADDER:**

Full Adder is the adder that adds three inputs and produces two outputs. The first two inputs are A and B and the third input is an input carry as C. The output carry is designated as Carry and the normal output is designated as S which is Sum.

Full Adder is designed by using previously implemented NAND and XOR gate.

Sum=A XOR B XOR C

Carry=A.B+B.C+C.A

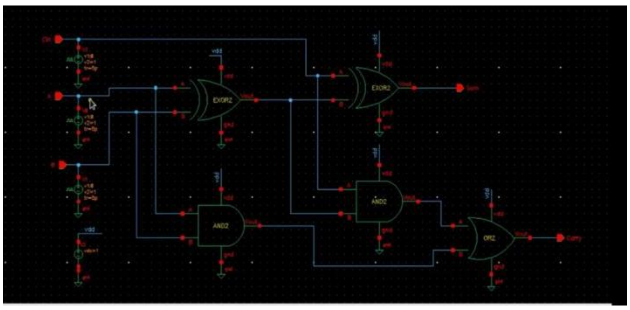

**Fig.7:** Schematic of Full Adder

**Fig.8:** Waveform of Full Adder

**CODE CONVERTER:**

Code Converters are the digital circuits or algorithms that are designed to translate data representation from one format to the other format.

$g0 = A \oplus B,$

$g1 = B \oplus C$

$g2 = C \oplus D$

$g3 = D$

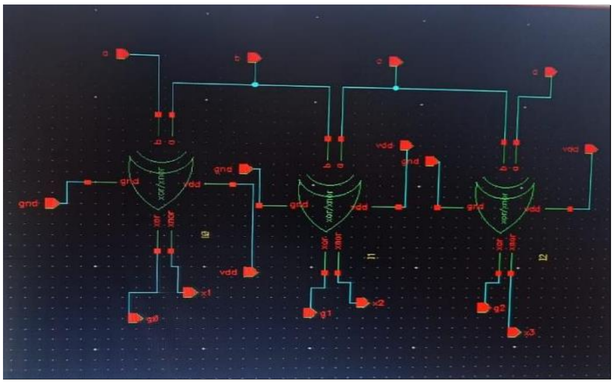

**Fig.9:** Schematic of Code Converter

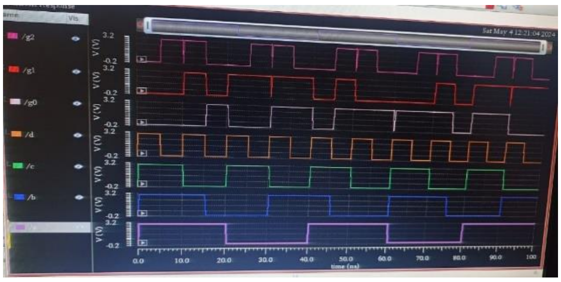

**Fig.10:** Waveform of Code Converter

**2x1 MUX:**

In 2×1 multiplexer, there are only two inputs, i.e., A0 and A1, 1 selection line, i.e., S0 and single outputs, i.e., Y. On the basis of the combination of inputs which are present at the selection line S0, one of these 2 inputs will be connected to the output.

$Y=\bar{S}B+SA$

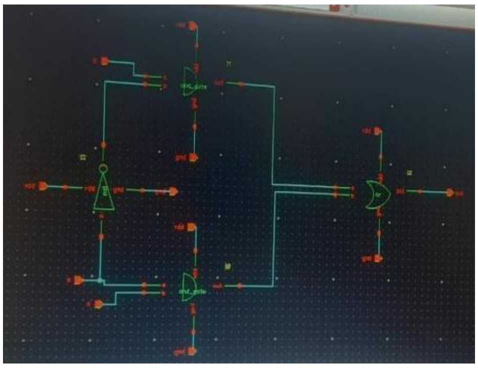

**Fig.11:** Schematic of 2x1 Mux

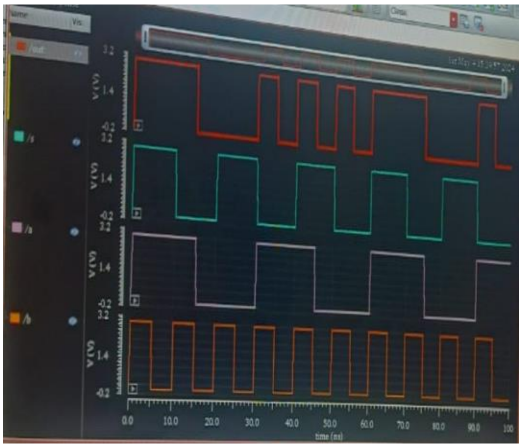

**Fig.12:** Waveform of 2x1 Mux

**4x1 MUX:**

A 4x1 multiplexer takes 4 inputs and directs a single selected input to output. The selection of input is controlled by selection inputs.

Y=S1′S0′I0+S1′S0I1+S1S0′I2+S1S0I3

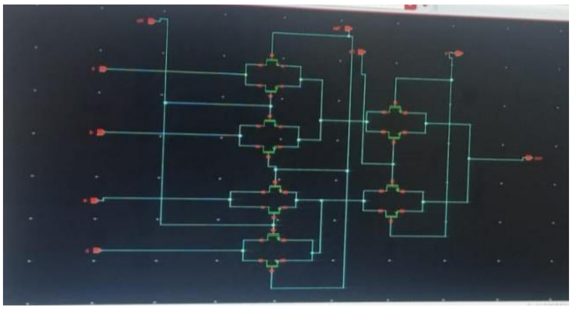

**Fig.13:** Schematic of 4X1 MUX

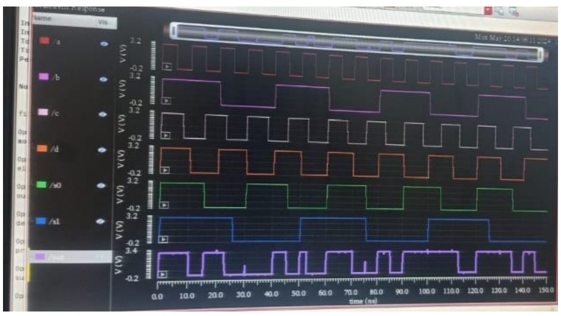

**Fig.14:** Waveform of 4X1 Mux

**4X1 MUX Using Adder, Code Converter, OR gate, NAND gate**

4-bit ALU is designed by using adder, NAND gate ,OR gate, Code Converter.

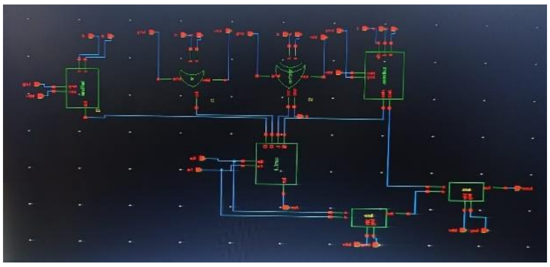

**Fig.15:** Schematic of 4X1 MUX using adder, Code converter ,Nor Gate, NAND gate

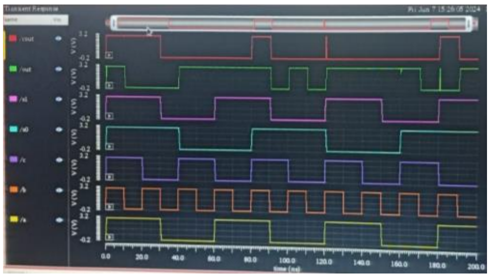

**Fig.16:** Waveform of 4X1 MUX using adder, Code converter , NOR Gate, NAND gate

| S1 | S0 | OUTPUT |
|---|---|---|
| 0 | 0 | NAND |
| 0 | 1 | Code Converter |
| 1 | 0 | OR |
| 1 | 1 | Adder |

**Table.1:** 4X1 MUX Truth Table for adder, Code converter , NOR Gate, NAND gate

**4X1 MUX Using XOR, XNOR , 1'S Compliment, NOR gate**

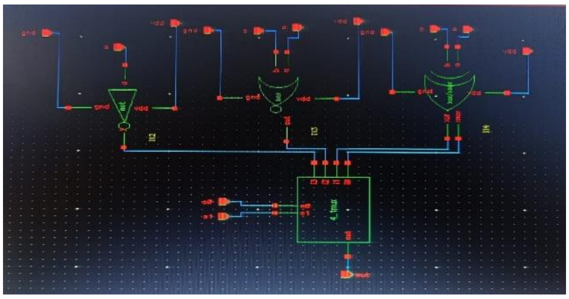

**Fig.17:** Schematic of XOR, XNOR, 1'S Compliment, NOR gate

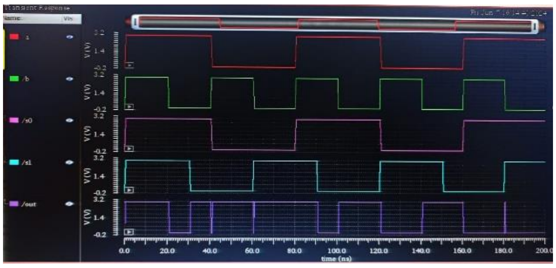

**Fig.18:** Waveform of XOR, XNOR , 1'S Compliment, NOR gate

| S1 | S0 | OUTPUT |
|---|---|---|
| 0 | 0 | 1'S Compliment |
| 0 | 1 | XOR |
| 1 | 0 | NOR |
| 1 | 1 | XNOR |

**Table.2:**4X1 MUX Truth Table for XOR, XNOR , 1'S Compliment, NOR gate.

Combing Above two 4x1 MUX by 2x1 MUX 1-bit 8-input ALU can be obtained as shown below:

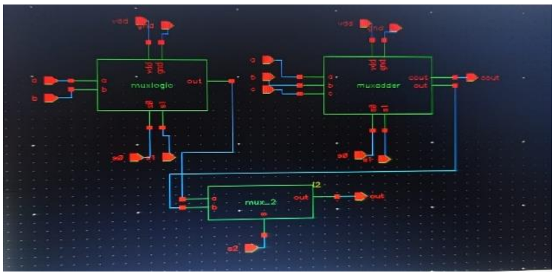

Fig.19: Schematic of 1-bit 8 input ALU

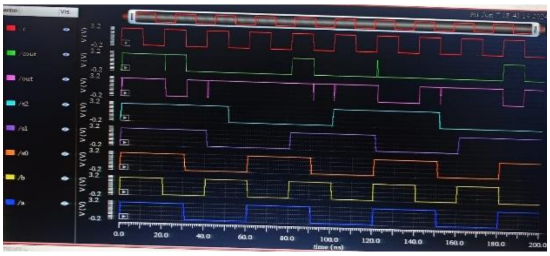

**Fig.20:** Waveform of 1-bit 8 input ALU

**4 bit ALU design from 1 bit ALU**

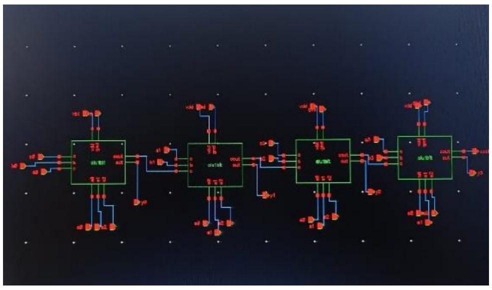

**Fig.21**: Schematic of 4 bit ALU

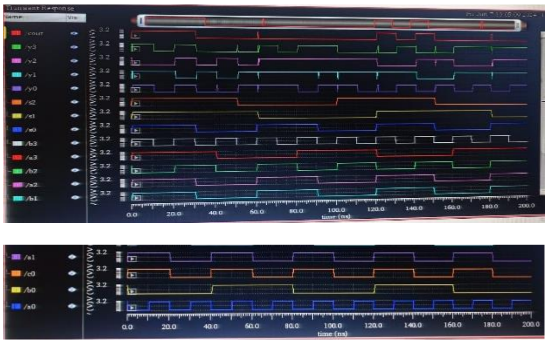

**Fig.22:** Waveform of 4 bit ALU

| S2 | S1 | S0 | OUTPUT |
|---|---|---|---|
| 0 | 0 | 0 | 1'S Compliment |
| 0 | 0 | 1 | XOR |
| 0 | 1 | 0 | NOR |
| 0 | 1 | 1 | XNOR |
| 1 | 0 | 0 | NAND |
| 1 | 0 | 1 | Code Converter |
| 1 | 1 | 0 | OR |
| 1 | 1 | 1 | Adder |

**Table.3:** 4-bit ALU Truth Table

## IV. Conclusion:

The design of the 4-bit ALU have been designed by using the sub blocks of full Adder, 4x1 and 2x1 multiplexers, and gates like exclusive-or, and, or and inverter. The ALU performs a total of eight operations Addition, Code Converter, NAND, NOR, XOR, XNOR, 1'S Complement, Comparator. The power is reduced compared with the existing ALU and power of the design is observed to be 4.23E-3. Hence the power is reduced.

The proposed design can be tested for low on chip power density further. Though we have utilized the power source efficiently, power can further be reduced by more efficiently utilizing the pulsed power supply as VDD source. Further the performance of proposed schematic may be increased by checking noise margin and current at different nodes.

## REFERENCE


[1] Low Power-Area Design Of Four Bit Arithmetic And Logical Unit (Alu) In Cadence Virtuoso Platform.

[2] D. Radhakrishnan,"Low-voltage low-power CMOS full adder", IEE Proceedings-Circuits, Devices and Systems, vol.148, pp.

[3] Raja l, Thanushkodi K, Hemalatha T (2023) ―Comparative Analysis of Various Low Power clock Gating Design for ALU,International Conference on Electronics and Communication, Vol 3, pp. 1-5.

[4] SUNG-MO KANG, YUSUF LEBLEBICI, "CMOS Digital Integrated Circuits Analysis and Design" 3rd Edition [5] https://www.semanticscholar.org/paper/Design-and-Implementation-of-4-Bit-Arithmetic-and-Yadav-Kumar/9e2deb0d787a8727b0e7b2d20ae2247cb6551eb2

[5] https://ieeexplore.ieee.org/document/8073484

[6] J.M. Wang, S.C. Fang and W.C. Fang, "New efficient designs for XOR and XNOR functions on transistor level", IEEE J. of Solid State Circuits, vol. 29, pp. 780-786, July 2020.

[7] Kaushik Roy and Sharat C Prasad, "Low power CMOS VLSI circuit design"2021

[8] Biswarup Mukherjee, Aniruddha Ghosal, "Design & Study of a Low Power High Speed Full Adder Using GDI Multiplexer" 2021 IEEE 2nd International Conference on Recent Trends in Information Systems (ReTIS).